Without conceding the examiner's position, the applicant proposes the claim amendments below. The applicant reserves the right to pursue any cancelled or amended claims in a continuing application.

1. (Currently amended) A method comprising:

one processor determining at least one queue parameter for a ~~first~~ process running on a ~~system~~ second processor; ~~and~~

~~using a queue management process separate from the first process to configure~~ configuring one or more queues on a storage device in accordance with the at least one queue parameter; and

carrying, on a bus between the processors, a flag indicating an operational condition of one of the queues used by the process.

2. (Previously presented) The method of claim 1 wherein said configuring one or more queues includes specifying a next read address indicative of a memory location within the storage device from which the next queue object requested from the queue is to be read.

3. (Previously presented) The method of claim 1 wherein said configuring one or more queues includes specifying a next write address indicative of a memory location within the storage device to which the next queue object provided to the queue is to be written.

4. (Previously presented) The method of claim 1 wherein said configuring one or more queues includes providing a queue status flag that is indicative of an operational condition of one of the queues .

5. (Original) The method of claim 1 wherein said configuring one or more queues includes specifying a starting address for the queue.

6. (Original) The method of claim 1 wherein the at least one queue parameter includes a queue depth parameter and said configuring one or more queues includes configuring the queue in accordance with the queue depth parameter.

7. (Original) The method of claim 1 wherein the at least one queue parameter includes a queue entry size parameter and said configuring one or more queues includes configuring the queue in accordance with the queue entry size parameter.

8. (Currently amended) A system comprising:

a host processor to determine at least one queue parameter for a ~~first~~ process running on said system;

a storage device; ~~and~~

a queue management process ~~separate from the first process~~ to configure one or more queues on said storage device in accordance with said at least one queue parameter; and

a flag bus for connecting said host processor to at least one slave processor.

9. (Previously presented) The system of claim 8 wherein said queue management process includes a read pointer process for each queue configured by said queue management process, wherein said read pointer process specifies a next read address indicative of a memory location within said storage device from which the next queue object requested from said queue is to be read.

10. (Previously presented) The system of claim 8 wherein said queue management process includes a write pointer process for each queue configured by said queue management process, wherein said write pointer process specifies a next write address indicative of a memory location within said storage device to which the next queue object provided to said queue is to be written.

11. (Currently amended) The system of claim 8 further comprising the at least one slave processor.

12. (Original) The system of claim 11 wherein said slave processor comprises a programmable state machine.

13. (Original) The system of claim 11 further comprising a data bus for connecting said host and slave processors, wherein said data bus transfers queue objects between said processors.

14. (Cancelled) The system of claim 11 further comprising a flag bus for connecting said host and slave processors.

15. (Original) The system of claim 14 wherein said queue management process includes a queue status monitoring process for each queue configured by said queue management process, wherein said queue status monitoring process provides a queue status flag, which is indicative of an operational condition of said queue, on said flag bus.

16. (Previously presented) The system of claim 15 wherein said queue status flag indicates at least one of:

an underflow queue condition;
an empty queue condition;
a nearly empty queue condition;
a nearly full queue condition;
a full queue condition; and
an overflow queue condition.

17. (Original) The system of claim 8 wherein said queue management process includes a queue base address process for each queue configured by said queue

management process, wherein said queue base address process specifies a starting address for said queue.

18. (Original) The system of claim 8 wherein said at least one queue parameter includes a queue depth parameter and said queue management process includes a queue depth specification process for each queue configured by said queue management process, wherein said queue depth specification process configures said queue in accordance with said queue depth parameter.

19. (Original) The system of claim 8 wherein said at least one queue parameter includes a queue entry size parameter and said queue management process includes a queue entry size specification process for each queue configured by said queue management process, wherein said queue entry size specification process configures said queue in accordance with said queue entry size parameter.

20. (Original) The system of claim 8 wherein said storage device comprises an SRAM storage device.

21. (Original) The system of claim 8 wherein said one or more queues temporarily store queue objects and said queue objects include at least one of:

a data packet; and

a system command.

22. (Currently amended) A computer program product residing on a computer readable medium having instructions stored thereon that, when executed by ~~the processor~~ a computer system, ~~cause that processor~~ causes the computer system to:

use one processor to determine at least one queue parameter for a ~~first~~ process running on a ~~system~~ second processor; ~~and~~

~~use a queue management process separate from the first process to~~ configure one or more queues on a storage device in accordance with the at least one queue parameter; and

carry, on a bus between the processors, a flag indicating an operational condition of one of the queues used by the process.

23. (Original) The computer program product of claim 22 wherein said computer readable medium comprises a read-only memory.

24. (Original) The computer program product of claim 22 wherein said computer readable medium comprises a hard disk drive.

25. (Currently amended) ~~A queue management process~~ An apparatus for configuring one or more queues, comprising:

a queue base address process for specifying a starting address for each of said one or more queues required by a process running on a system; and

a queue depth specification process, working in conjunction with the queue base address process, for configuring each said queue in accordance with a queue depth parameter provided by said process running on said system.

26. (Currently amended) The ~~queue management process~~ apparatus of claim 25 further comprising:

a queue entry size specification process for configuring each said queue in accordance with a queue entry size parameter provided by said process running on said system.

27. (Currently amended) ~~A queue management process~~ An apparatus for configuring one or more queues, comprising:

a queue base address process for specifying a starting address for each of said one or more queues required by a process running on a system; and

a queue entry size specification process, working in conjunction with the queue base address process, for configuring each said queue in accordance with a queue entry size parameter provided by said process running on said system.

28. (Currently amended) The ~~queue management process~~ apparatus of claim 27 further comprising:

a queue depth specification process for configuring each said queue in accordance with a queue depth parameter provided by said process running on said system.

29. (Currently amended) ~~A queue management process~~ An apparatus for configuring one or more queues, comprising:

a queue base address process for specifying a starting address for each of said one or more queues required by a process running on a system; and

a queue status monitoring process, working in conjunction with the queue base address process, for providing, for each said queue, a queue status flag that is indicative of the operational condition of said queue.

30. (Currently amended) The ~~queue management process~~ apparatus of claim 29 wherein said queue status flag is configured to indicate at least one of:

an underflow queue condition;

an empty queue condition;

a nearly empty queue condition;

a nearly full queue condition;

a full queue condition; and

an overflow queue condition.

31. (Currently amended) Circuitry ~~to determine at least one queue parameter for a first process to be run on a system, said at least one queue parameter to be used to configure one or more queues, said circuitry~~ comprising:

a first processor to determine at least one queue parameter for a process to be run on a second processor;

a storage device; and

a queue management process ~~separate from the first process~~ to configure ~~said one~~ one or more queues on said storage device in accordance with said at least one queue parameter; and

a bus, connecting the processors, to carry a flag indicating an operational condition of one of the queues.

32. (Previously presented) The circuitry of claim 31 wherein said queue management process includes a read pointer process for each queue configured by said queue management process, wherein said read pointer process is configured to specify a next read address indicative of a memory location within said storage device from which the next queue object requested from said queue is to be read.

33. (Previously presented) The circuitry of claim 31 wherein said queue management process includes a write pointer process for each queue configured by said queue management process, wherein said write pointer process is configured to specify a next write address indicative of a memory location within said storage device to which the next queue object provided to said queue is to be written.

34. (Currently amended) The circuitry of claim 31 ~~further comprising~~ in which the first and second processors comprise a host processor and ~~at least one~~ a slave processor.

35. (Original) The circuitry of claim 34 further comprising a data bus for connecting said host and slave processors, wherein said data bus transfers queue objects between said processors.

36. (Cancelled) The circuitry of claim 34 further comprising a flag bus for connecting said host and slave processors.

37. (Original) The circuitry of claim 36 wherein said queue management process includes a queue status monitoring process for each queue configured by said queue management process, wherein said queue status monitoring process provides a queue status flag, which is indicative of an operational condition of said queue, on said flag bus.

38. (Original) The circuitry of claim 31 wherein said queue management process includes a queue base address process for each queue configured by said queue

management process, wherein said queue base address process specifies a starting address for said queue.

39. (Original) The circuitry of claim 31 wherein said at least one queue parameter includes a queue depth parameter and said queue management process includes a queue depth specification process for each queue configured by said queue management process, wherein said queue depth specification process configures said queue in accordance with said queue depth parameter.

40. (Original) The circuitry of claim 31 wherein said at least one queue parameter includes a queue entry size parameter and said queue management process includes a queue entry size specification process for each queue configured by said queue management process, wherein said queue entry size specification process configures said queue in accordance with said queue entry size parameter.